

MODELING TEMPORAL INFORMATION USING DISCRETE FOURIER TRANSFORM FOR RECOGNIZING EMOTIONS IN USER-GENERATED VIDEOS

Haimin Zhang and Min Xu

GBDTC, Faculty of Engineering and IT, University of Technology, Sydney
Haimin.Zhang@student.uts.edu.au, Min.Xu@uts.edu.au

ABSTRACT

With the widespread of user-generated Internet videos, emotion recognition in those videos attracts increasing research efforts. However, most existing works are based on frame-level visual features and/or audio features, which might fail to model the temporal information, *e.g.* characteristics accumulated along time. In order to capture video temporal information, in this paper, we propose to analyse features in frequency domain transformed by discrete Fourier transform (DFT features). Frame-level features are firstly extract by a pre-trained deep convolutional neural network (CNN). Then, time domain features are transferred and interpolated into DFT features. CNN and DFT features are further encoded and fused for emotion classification. By this way, static image features extracted from a pre-trained deep CNN and temporal information represented by DFT features are jointly considered for video emotion recognition. Experimental results demonstrate that combining DFT features can effectively capture temporal information and therefore improve emotion recognition performance. Our approach has achieved a state-of-the-art performance on the largest video emotion dataset (VideoEmotion-8 dataset), improving accuracy from 51.1% to 55.6%.

Index Terms— video emotion recognition, discrete Fourier transform, convolutional neural network

1. INTRODUCTION

With the advances of mobile devices with high quality cameras, people can easily capture, edit and upload video clips to the Internet. It is urgent to develop intelligent video search engines. Rather than semantic based video searching, people may prefer to pick up their favourite videos according to their emotional preferences since videos may convey different kinds of emotions. Therefore, how to automatically recognise the emotions conveyed in these videos has attracted increasing interests in the research community.

While dramatic progress has been achieved in affective image classification [1, 2, 3], few attempts have been made to video emotion recognition. Compared to affective image analysis, video emotion analysis is a complex task. This is

mainly because of two reasons: 1) Different from a single image, a video sequence contains multiple frames which group together to reflect a dominant emotion. Within a video sequence, some frames might not convey (or even convey an opposite emotion of) the dominant emotion. 2) As one of the temporal features, characteristics accumulated along time, is difficult to model. Early studies on video emotion recognition mainly focused on movie videos [4, 5]. Most of these studies utilised low-level visual features (*e.g.* colour and brightness) and/or audio features (*e.g.* MFCC and Audio-Six). In addition to low-level features, Jiang *et. al.* [6] introduced so-called attribute features (*e.g.* ObjectBank [7] and Sentibank [3]) for emotion recognition.

In contrast to hand-crafted features, the last few years has witnessed the success of deep features. Deep features extracted from the activation of a convolutional neural work (CNN) pre-trained on ImageNet [8] are more discriminative than hand-crafted features [9]. CNN features have achieved state-of-the-art results on many benchmarks [10] and therefore are widely used in image classification, object detection and attribute detection [11, 10]. Recently, researchers started applying CNN features to video classification [12, 13, 14]. For example, Xu *et. al.* [14] proposed a video representation by leveraging frame-level features extracted by CNN with an appropriate feature encoding method.

Most conventional approaches for video classification involve mainly three stages: First, local features or frame-level features are extracted. Then these features are quantised to a fixed length representation using a visual dictionary usually learned by K-means algorithm. Later, video-level representations are obtained by pooling method, such as max-pooling and average-pooling. Lastly, a classifier is trained on the video-level representations to differentiate different classes of video emotions.

However, during the process of feature quantisation and pooling, the temporal information (*i.e.*, information accumulated along time) of videos has not been well treated. Most recently, realising the importance of temporal information for video classification, researchers have started exploring how to take temporal information into account for video analysis. Lan *et. al.* [15] proposed to model temporal information using temporal division pyramid (TDP) for action recognition.

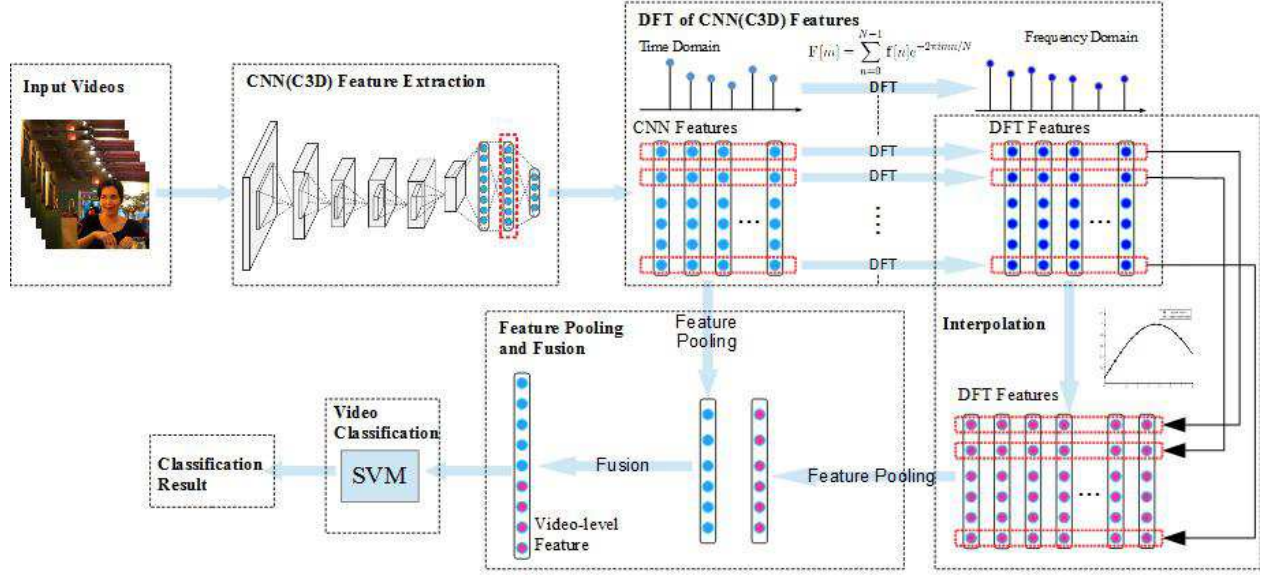


Fig. 1. An overview of our method for video emotion classification.

However, the dimension of the video-level features generated by TDP method are very high, which might lead to a problem of classifier overfitting. In [13], Long Short Term Memory (LSTM) [16], which can preserve information for a long time, was adopted to model temporal information for video classification. However, training an LSTM is a time-consuming task.

Inspired by DFT, which can transform a discrete signal from time domain to frequency domain, this paper proposes to analysis features in frequency domain to model temporal information for videos. To some extent, signal characteristics along time can be accumulated and represented through sampling in frequency domain. In this work, we integrate CNN features and DFT features for video representation by leveraging feature encoding method.

2. OVERALL ARCHITECTURE

The proposed method consists of five steps as shown in figure 1. The first step is to extract CNN features from video frames. Secondly, considering each feature dimension as a discrete signal over time, we apply DFT to transform the signal to frequency domain. Thirdly, interpolation method is adopted to generate a fixed length representation for every dimension of DFT features. Fourthly, locality-constrained linear coding (LLC) method and max-pooling strategy are applied to aggregate CNN features and DFT features. The combination of the aggregated CNN features and DFT features can be regarded as a video-level representation. Finally, with the features generated in the fourth step, an SVM is trained to recognise emo-

tions. Different from existing methods for temporal information analysis, our method has the following advantages:

- Compared to LSTM, our method avoids a complex model learning process.
- Our method captures both spatial information and temporal information with relatively low feature dimensions.
- Video clips have different length which forms a difficulty for forming a uniformed feature representation. Using DFT is resilient to signal length variation. Moreover, with fast Fourier transform (FFT), computational complexity can be significantly reduced.

The rest of this paper is organised as follows. In section 3, the proposed approach is introduced in detail. Experimental results and comparison with previous works are present in section 4. Finally, we conclude this paper in section 5.

3. THE PROPOSED ALGORITHM

In this section, the proposed algorithm is introduced in detail step by step.

3.1. Feature Extraction

As shown in [10], deep features extracted from a convolutional neural network which is pre-trained on a large image dataset can be used as a powerful feature representation for many visual analysis tasks. In this paper, we leverage a

pre-trained deep convolutional neural network [17] to extract frame-level descriptors for all video clips. The network consists of five convolution layers and three fully connected layers with a final 1000-way softmax. All input images are resized to 256×256 without considering its original aspect ratio before feeding into the network. Considering features extracted from fully connected layers can capture semantic information for the input image, activation from the last fully connected layer (referred to as fc_7 [14]) are extracted as the frame descriptor. Then ℓ_2 normalization is adopted for all frame-level descriptors. Let \mathbf{f} denote the set of frame-level descriptors of a video clip which has N frames, then \mathbf{f} can be described as

$$\mathbf{f} = (\mathbf{f}_1, \dots, \mathbf{f}_N) = \begin{bmatrix} \mathbf{f}_1[1] & \dots & \mathbf{f}_i[1] & \dots & \mathbf{f}_N[1] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{f}_1[k] & \dots & \mathbf{f}_i[k] & \dots & \mathbf{f}_N[k] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{f}_1[D] & \dots & \mathbf{f}_i[D] & \dots & \mathbf{f}_N[D] \end{bmatrix}$$

Where $\mathbf{f}_i = (\mathbf{f}_i[1], \dots, \mathbf{f}_i[D])^T$ represents the descriptor of the i -th frame with dimension D . In this work, D equals 4096, which is the dimension of fc_7 . The value of N can be different for different video clips.

3.2. Discrete Fourier Transform of CNN Features

The aim of discrete Fourier transform (DFT), which is widely used in the field of signal processing, is to transform a discrete signal from time domain to frequency domain. At this step, we present how DFT is applied to CNN features.

As described in 3.1, $\mathbf{f} = (\mathbf{f}_1, \dots, \mathbf{f}_N)$ represents the set of frame-level CNN features extracted from a video clip with N frames. Let the k -th dimension of \mathbf{f} be denoted $\mathbf{f}[k] = (\mathbf{f}_1[k], \mathbf{f}_2[k], \dots, \mathbf{f}_N[k])$. $\mathbf{f}[k]$ can be considered as a discrete signal which has N sample points with equal sampling time intervals Δt . We transform $\mathbf{f}[k]$ to frequency domain using the following equation

$$\mathbf{F}_s[k] = \sum_{n=1}^N \mathbf{f}_n[k] e^{-2i\pi(n-1)(s-1)/N}, s = 1, 2, \dots, N. \quad (1)$$

Let the result be denoted $\mathbf{F}[k] = (\mathbf{F}_1[k], \dots, \mathbf{F}_N[k])$, the number of points obtained in frequency domain is as same as that in time domain. The computed value $\mathbf{F}_s[k] \in \mathbf{F}[k]$ is a complex number and its absolute value represents the amplitude of the s -th frequency. In our work, the absolute value of $\mathbf{F}_s[k]$ is used instead of its original complex value. After transforming all $\mathbf{f}[k]$ to frequency domain, where $k =$

$1, \dots, D$, we get the following feature set

$$\mathbf{F} = (\mathbf{F}_1, \dots, \mathbf{F}_N) = \begin{bmatrix} \mathbf{F}_1[1] & \dots & \mathbf{F}_i[1] & \dots & \mathbf{F}_N[1] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{F}_1[k] & \dots & \mathbf{F}_i[k] & \dots & \mathbf{F}_N[k] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{F}_1[D] & \dots & \mathbf{F}_i[D] & \dots & \mathbf{F}_N[D] \end{bmatrix}$$

Where $\mathbf{F}_i = (\mathbf{F}_i[1], \dots, \mathbf{F}_i[D])^T$, termed as a DFT feature in this paper.

3.3. Interpolation

As mentioned in section 3.1, the number of sample points N is different due to the various video length. For two video clips u and v with N and M frames respectively, let $\mathbf{f}^u[k] = (\mathbf{f}_1^u[k], \mathbf{f}_2^u[k], \dots, \mathbf{f}_N^u[k])$ and $\mathbf{f}^v[k] = (\mathbf{f}_1^v[k], \mathbf{f}_2^v[k], \dots, \mathbf{f}_M^v[k])$ indicate the k -th dimension of CNN features. We use Δt to indicate the sampling time interval which is uniform for all video clips, *i.e.* the sampling rate is $S = 1/\Delta t$.

After transforming $\mathbf{f}^u[k]$ and $\mathbf{f}^v[k]$ to frequency domain, we obtain $\mathbf{F}^u[k] = (\mathbf{F}_1^u[k], \mathbf{F}_2^u[k], \dots, \mathbf{F}_M^u[k])$ and $\mathbf{F}^v[k] = (\mathbf{F}_1^v[k], \mathbf{F}_2^v[k], \dots, \mathbf{F}_M^v[k])$ respectively. $\mathbf{F}^u[k]$ and $\mathbf{F}^v[k]$ have the same frequency range from 0 to S with sampling interval S/N and S/M respectively. From equation 1, we know that the number of points obtained in frequency domain is as same as that in time domain. Therefore signals with more sample points in time domain are more compactly spaced in frequency domain than signals with less sample points.

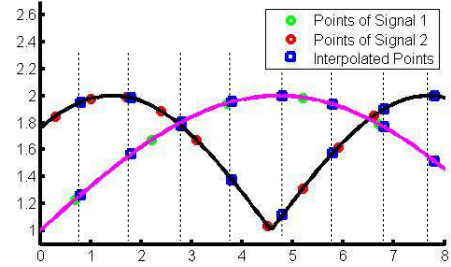


Fig. 2. Illustration of applying interpolation to signals with different points in frequency domain.

Like image resizing, we use cubic interpolation method [18] to generate a fixed length representation $\mathbf{F}^u[k] = (\mathbf{F}_1^u[k], \mathbf{F}_2^u[k], \dots, \mathbf{F}_L^u[k])$ and $\mathbf{F}^v[k] = (\mathbf{F}_1^v[k], \mathbf{F}_2^v[k], \dots, \mathbf{F}_L^v[k])$. By this way, different video clips have the same frequency sample interval from the frequency range from 0 to S , as shown in Figure 2.

3.4. Feature Encoding and Fusion

In order to generate a uniform video-level representation, we need to aggregate the obtained CNN features and DFT fea-

tures separately. Firstly, locality-constrained linear coding (LLC) algorithm [19] is adopted for feature embedding. LLC is an extension of the sparse coding method. It utilises locality constraints, which selects k-nearest codewords from a dictionary learned by K-means algorithm, and generates a sparse representation for the input vector. After that, we choose to use max-pooling strategy to aggregate the LLC-based features.

We then adopt late feature fusion. The linear combination of the aggregated LLC-based CNN features and DFT features lead to the final video-level representation.

3.5. Emotion Classification

After obtaining all video-level features, an SVM is trained by optimising the following equation [20] for emotion classification.

$$\arg \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^T \max(1 - y_i(\mathbf{w}^T \mathbf{x}_i) + b), 0) \quad (2)$$

Where \mathbf{x}_i and y_i represent video-level feature and its corresponding label, C and T indicate penalty parameter and the number of training features respectively.

4. EXPERIMENTS

4.1. Experimental Setup

Experiments were conducted on the VideoEmotion-8 dataset [6], which contains 1,101 user-generated videos labelled with 8 basic human emotion categories. Each category is with a minimum number of 100 videos per class and an average duration of 107 seconds. Currently, this dataset is the largest dataset for recognising emotions in user-generated videos.

Similar as [6], we ran test for 10 times. For each time, we randomly selected 2/3 data from each class for training and the rest for testing. The average accuracy was further calculated to evaluate the classification performance.

For computation efficiency, we sampled a frame every 8 frames. Frame-level descriptors were extracted using the Caffe toolkit [9]. We selected 50 videos randomly from each emotion category for fine-tuning. The CNN model was first pre-trained on ImageNet [8] and then fine-tuned by using the video frames. All frame-level descriptors were further ℓ_2 normalised. Fast Fourier transform (FFT) [21] was adopted for computing DFT. At the interpolation step, L was set to 500 experimentally. For LLC coding, we trained a vocabulary with 1024 codewords for CNN features and DFT features separately. The aggregated LLC-based CNN features and DFT features were normalised to 3/5 and 2/5 respectively using ℓ_2 normalisation.

We applied the LibLinear toolbox [20] for SVM classification. The penalty parameter C was set to 1 experimentally.

Category	Jiang [6]	Pang [22]	CNN	DFT	CNN+DFT
Anger	53.0	48.5	58.6	53.7	62.1
Anticipation	7.6	0	41.6	32.7	42.4
Disgust	44.6	53.8	40.1	44.3	50.3
Fear	47.3	52.7	50.7	44.0	51.8
Joy	48.3	54.2	45.1	46.7	53.7
Sadness	20.0	32.4	60.0	43.1	60.9
Surprise	76.9	78.7	74.4	66.0	76.2
Trust	28.5	43.8	47.8	24.6	47.3
Overall	46.1	51.1	52.3	44.4	55.6

Table 1. Prediction accuracy (%) of each emotion category using CNN, DFT and the concatenation of the two features, and comparison with previous works.

4.2. Experimental Results and Discussion

In our experiments, we intended to evaluate: (1) the performance of CNN features; (2) the performance of DFT features; and (3) the overall performance of combined CNN and DFT features. Moreover, to prove the efficiency of our proposed method, we compared our results with the most recent two works [6] and [22]. Experimental results are shown in Table 1, from which we can find:

- (1) In comparison with [6] and [22], the accuracy of our method using CNN features improves 6.2% and 1.2% respectively. In [6] and [22], the authors used low-level visual features, audio features and attribute features, whereas CNN features were applied in our work. Experimental results demonstrate that the performance of CNN features may be superior than hand-crafted features.
- (2) Using DFT features only can not improve the classification performance.
- (3) Combining CNN and DFT features, our method outperforms [6] 9.5% and [22] 4.5%. Overall, incorporating DFT features, the classification performance can be improved 3.3% compared with applying CNN feature only. The results prove the efficiency of the DFT features. The major reason might be that we modeled temporal information by analysing DFT features. By combining temporal information, the recognition accuracy can be improved to some extent. Another possible reason is that LLC was used for feature encoding, which might had better performance than bag of words model that was used in [6]. To the best of our knowledge, this is the state-of-the-art performance on VideoEmotion-8 dataset.

5. CONCLUSIONS

In this paper, we have proposed to apply CNN and DFT features to model spatial and temporal information for video emotion analysis. Compared to hand-crafted features, CNN features can improve the classification accuracy significantly. Capturing temporal information, DFT features have been further proved to be efficient and effective. The combination of CNN and DFT features achieves the state-of-the-art performance on VideoEmotion-8 dataset.

6. REFERENCES

- [1] Jana Machajdik and Allan Hanbury, "Affective image classification using features inspired by psychology and art theory," in *Proceedings of the international conference on Multimedia*. ACM, 2010, pp. 83–92.
- [2] Ningning Liu, Emmanuel Dellandrea, Bruno Tellez, and Liming Chen, "Associating textual features with visual ones to improve affective image classification," in *Affective Computing and Intelligent Interaction*, pp. 195–204. Springer, 2011.
- [3] Damian Borth, Tao Chen, Rongrong Ji, and Shih-Fu Chang, "Sentibank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content," in *Proceedings of the 21st ACM international conference on Multimedia*. ACM, 2013, pp. 459–460.
- [4] Hee Lin Wang and Loong-Fah Cheong, "Affective understanding in film," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 16, no. 6, pp. 689–704, 2006.
- [5] Min Xu, Changsheng Xu, Xiangjian He, Jesse S Jin, Suhui Luo, and Yong Rui, "Hierarchical affective content analysis in arousal and valence dimensions," *Signal Processing*, vol. 93, no. 8, pp. 2140–2150, 2013.
- [6] Yu-gang Jiang, Baohan Xu, and Xiangyang Xue, "Predicting Emotions in User-Generated Videos," *International Conference on Artificial Intelligence (AAAI)*, pp. 73–79, 2014.
- [7] Li-Jia Li, Hao Su, Yongwhan Lim, and Li Fei-Fei, "Object bank: An object-level image representation for high-level visual recognition," *International journal of computer vision*, vol. 107, no. 1, pp. 20–39, 2014.
- [8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on CVPR, 2009*. IEEE, 2009, pp. 248–255.
- [9] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.
- [10] Ali S Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. IEEE, 2014, pp. 512–519.
- [11] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, "Learning deep features for scene recognition using places database," in *Advances in Neural Information Processing Systems*, 2014, pp. 487–495.
- [12] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.
- [13] Zuxuan Wu, Xi Wang, Yu-Gang Jiang, Hao Ye, and Xiangyang Xue, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," *arXiv preprint arXiv:1504.01561*, 2015.
- [14] Zhongwen Xu, Yi Yang, and Alexander G Hauptmann, "A discriminative cnn video representation for event detection," *arXiv preprint arXiv:1411.4006*, 2014.
- [15] Zhenzhong Lan, Xuanchong Li, and Alexander G Hauptmann, "Temporal extension of scale pyramid and spatial pyramid matching for action recognition," *arXiv preprint arXiv:1408.7071*, 2014.
- [16] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [18] Robert G Keys, "Cubic convolution interpolation for digital image processing," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 29, no. 6, pp. 1153–1160, 1981.
- [19] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong, "Locality-constrained linear coding for image classification," in *IEEE Conference on CVPR, 2010.*, pp. 3360–3367.

- [20] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, “Liblinear: A library for large linear classification,” *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.
- [21] Matteo Frigo and Steven G Johnson, “Fftw: An adaptive software architecture for the fft,” in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*. IEEE, 1998, vol. 3, pp. 1381–1384.
- [22] Lei Pang, Shiai Zhu, and Chong-Wah Ngo, “Deep multimodal learning for affective analysis and retrieval,” *Multimedia, IEEE Transactions on*, vol. 17, no. 11, pp. 2008–2020, 2015.